2017

[IBM GURUKOOL]

Project Scenarios

Project Name:

Sales Data Tracker_Report Generation and View Reports

Table of Contents

1 Story:

There is an FMCG company based in Raipur. Its operation is spread across different states of India. The company sales its product through their salesperson, who goes door-to-door or shop-to-shop to sell the products. The Sale division of this company work in small teams. In a team there is one Team-Lead and 5 to 6 Team members, i.e. Sales-Persons. The team members report to the Team-Lead. They have to submit their daily sales transaction report to the Team-Lead at fixed time.

This company follows traditional way of reporting i.e. they report it over the phone. It becomes quite a daunting task for a company to manage all the reporting and then generating the required reports. It is also difficult for a company to do every time follow ups when any sales person has failed to submit its sales report. In order to remove all these flaws, the company wants to upgrade this process and plans to adopt online reporting and tracking of their sales data. For this they want to develop a sales data tracking software.

This full application is all about tracking of sales data. There are various modules involved in the development of this complete Project like Login, Sales Data Entry, Data Correction Module, Leave Management, Reporting, Reminder Service, View Report, Maintenance and Performance Report.

In a sales data tracking software every user has its own login ID. In this software user can be Sales Person, Team Lead, Super admin etc.
Sales person has got different functionality to use this software as to what a Lead has got and likewise for a Manager also.
Depending upon the type of user, all the functionalities are defined. Since this Software is for tracking of sales data so data entry becomes a vital role here. It is the sales person who submits their day to day sales transaction data. And if sales person has to do any modification they can also modify their last submitted data. If they fail to submit their report within their due time then an auto reminder mail go to the defaulter and if again s/he fail to submit their daily sales report then this time a reminder mail go to the defaulter along with the Team Lead.
This application also has provision where if any sales person is on leave then s/he can delegate other person on her/his behalf to submit their sales entry. When all the data are submitted then reports will be either auto-generated or generated manually by higher authorities and then it is analysed completely to collect all the information.
The reports can be forwarded to other user also either manually or automatically. This also provides provision to user to see the performance of product, team and individual. This application also has a provision for report archival where user can see the old reports.
In this application there is a role of super admin who has the full responsibility of maintaining this entire application.

This entire project is divided into various sub modules which are listed below:-

- Login
- Sales Data Entry
- Data Correction module
- Report Forwarding
- Report Generation
- Report Dispatcher
- Reminder Service
- View Report
- Maintenance – User & Team
- Maintenance – Delegation
- Maintenance – Leave
- Maintenance – Verbs
- Maintenance - Services
- Maintenance – Products
- Performance Report
- Email Service
- Sales Forecaster

2 Database:

The Database Table Structure is described below:-

Database	dbSaleDataTracker

tUserInfromations			
	ID	int	
	FullName	varchar(200)	
	EMailAddress	varchar(200)	

tUserLoginCredentials			
	ID	int	
	UserID	int	tUserInfromations.ID
	LoginID	varchar(20)	
	Password	varchar(20)	

tSalesTeamInformation			
	ID	int	
	TeamName	varchar(20)	
	TeamLeadID	int	tUserInfromations.ID
	CreatedDate	date	

tUserReportingHierarchy			
	ID	int	
	LeadsUserID	int	tSalesTeamInformation.TeamLeadID
	MembersUserID	int	tUserInfromations.ID

tUserReportingDelegation			
	ID	int	
	DelegatorsUserID	int	tUserInfromations.ID
	DelegateesUserID	int	tUserInfromations.ID
	DelegationStartDate	date	
	DelegationEndDate	date	

tUserLeaveInfromations			
	ID	int	
	UserID	int	tUserInfromations.ID
	LeaveStartDate	date	
	LeaveEndDate	date	

tProductInfromations			
	ID	int	
	Name	varchar(200)	
	Code	varchar(10)	

tProductSalesDataSubmitted			
	ID	int	
	SalesDataInputString	varchar(1000)	tProductInfromations.Code
	SubmittedBy	int	tUserInfromations.ID
	SubmittedFor	int	tUserInfromations.ID
	SubmittedDate	dateTime	

tProductSalesDataProcessed			
	ID	int	
	SubmitionID	int	tProductSalesDataSubmitted.ID
	ProductCode	varchar(10)	tProductInfromations.Code
	SalesCount	int	

tSalesReports			
	ID	int	
	RecipientID	int	tUserInfromations.ID
	ReportString	varchar(1000)	
	ReportDate	date	

tSalesReportsSendStatus			
	ID	int	
	SalesReportID	int	tSalesReports.ID
	RecipientID	int	tUserInfromations.ID
	SendDateTime	dateTime	

tSalesDataVerbs			
	ID	int	
	Verb	varchar(30)	DEL
	Description	varchar(30)	Delete
	Action	varchar(1000)	Delete given dates record

tReportingServiceKeyValueActions			
	ID	int	
	Key	varchar(30)	DAILY_REPORT_TIME
	Value	varchar(30)	2100
	Action	varchar(1000)	

tReminderServiceKeyValueActions			
	ID	int	
	Key	varchar(30)	DAILY_REMINDER_TIME
	Value	varchar(30)	2000
	Action	varchar(1000)	send remover at 8:00 pm

3 Project Selection:

The 'coolDataTracker' is divided in small-small modules. The modules are carved out in a way that they become a fully-independent, fully-functional project by themselves but when they all are integrated together, it will form fully-functional 'coolDataTracker'.

The students' team is required to choose one of the modules as their project and work on that module only.
Wherever required, use the above database structure only.
Note: Any change in database might lead to integration issues, therefore student are requested to take guide's permission before make any change in database structure.

4 Modules:

There are 17 modules in this application. Among these 17 modules, Report Generation and View Reports is one of them.

4.1 Report Generation

Objective:

- To have automated service will generate report at pre-defined time.
- The Generated Report should also show pictorially the sales report of a User. The report should be shown in bar chart and pie chart with color code.

  Color Code:-

  a) **Blue-** Above Average
  b) **Green-** Average
  c) **Red-** Below Average

- Along with the automated service, application should also have a web page to generate report manually.

Assumptions:

- There is super admin in database who will manually generate the Report.
- Format of the Report should be:
  LeadID Product Code Sales Number
  e.g. Lead1 Prod1 40
  Lead1 Prod2 50

Pre -Requisite:

- If database, required tables or data is not available, developer can create test data a per table structure
- There will be a key 'DAILY_REPORT_GENERATION_TIME' in table 'tReportingServiceKeyValueActions'

Workflow:

- Application will first find Submitted Sales Data. It should be in the Format:
  UserID ProductCode Sales Number ProductCode Sales Number
  E.g. SE1 Prod1 10 Prod 2 30
  SE2 Prod1 30 Prod2 20 (Assuming SE1 & SE2 is under Lead1)

- Application will then find Processed Sales Data from the Submitted Sales Data. It should be in the following Format:
  UserID Product Code Sales Number
  Eg SE1 Prod1 10
  SE1 Prod2 30
  SE2 Prod1 30
  SE2 Prod2 20
- Application will then find the Sales Report in the format given in assumption
- For Auto generation of the report, application will check tReportingServiceKeyValueActions.Value with the system clock. If it matches then autogenerate the report at that time.
- For manual generation of report there will be 1 button. Upon clicking of that button any time it just generate and display all the reports of the day.

Database Tables Involved:

- tProductSalesDataProcessed
- tSalesReports
- tReportingServiceKeyValueActions

4.2 View Reports

Objective:

To have web page where user can see the old reports. User can get report for a particular user, team, of a given date or between given dates.

Assumptions:

- There are database and tables ready.

Pre -Requisite:

If database, required tables or data is not available, developer can create test data a per table structure

Workflow:

- Application will find the reports based on User and Team.
- If any user wants to see the reports based on user then application should find and display all the reports of the selected user for a given date or between given dates. Validate if the username in string is present in database.
  a) If Yes then display the reports of that user.
  b) If No then reject the input and display appropriate message.
- If any user wants to see the reports based on Team then application should find and display all the reports of the

selected team for a given date or between given dates. Validate if the team name in string is present in database

a) If Yes then display all the results of that team.
b) If No then reject the input and display appropriate message.

Database Tables Involved:

- tUserInfromations
- tSalesTeamInformation
- tUserReportingHierarchy
- tSalesReports